

Les classes en JAVA

Date de dernière mise à jour : 27/06/2007 à 19:36

Source : <http://www.vulgarisation-informatique.com/classes-java.php>.

Distribution interdite sans accord écrit d'Anthony ROSSETTO (<http://www.vulgarisation-informatique.com/contact.php>)

JAVA est un langage entièrement orienté objet. Tout votre code doit donc être inclus dans ce qu'on appelle des classes (il vous en faudra au moins une quelle que soit l'application désirée). On décompose chaque classe en un fichier texte qui obéit aux [conventions de codage JAVA](#).

Les classes :

Une classe s'écrit comme ceci :

```
public
class
MaClasse{
//code
}
```

Il n'est pas nécessaire d'ajouter un constructeur à une classe. Un constructeur est une méthode qui sera appelée dès la création de l'objet (instanciation de la classe). Cependant, si votre classe n'a ni constructeur ni méthode main() (nous allons revenir sur cette méthode plus bas) et qu'il s'agit de l'unique classe de votre projet, vous ne pourrez rien faire de concret. C'est pourquoi dans chaque projet il vous faut au moins une méthode main() qui sera appelée lorsque vous utiliserez la commande **java NomClasse** à partir de la console de votre système d'exploitation. Voici l'exemple d'une classe avec un constructeur :

```
public
class
Test{
    public
    Test(
    )
    {
        System.out.println(
        &quot;L'objet a été créé&quot;;
    )
    ; }
}
```

Lorsque vous créerez l'objet Test via cette instruction : `Test variable = new Test();`, ceci affichera à l'écran `"L'objet a été créé"`. Ceci ne fonctionnera pourtant toujours pas si votre classe est la seule du projet et que vous souhaitez l'exécuter. Pour pouvoir exécuter votre classe sans l'intermédiaire d'un environnement de développement intégré (aussi appelé IDE) comme par exemple [BlueJ](#), vous devez impérativement ajouter une méthode main() à votre classe.

Méthode main() :

```
public
class
Test{
    public
    static
    void
```

```
main(  
String[] args)  
{  
    System.out.println(  
&quot;Le projet a été lancé&quot;  
)  
;  
}
```

Comme vous pouvez le voir, nous avons utilisé quelques mots inconnus jusqu'alors. Parmi ces mots, on remarque la présence des mots clés **static** et **void** que vous aurez l'occasion de voir dans la section [JAVA](#) du site.

Le mot clé static signifie que la méthode main() est statique, c'est à dire que vous pouvez y accéder sans avoir créé d'instance de l'objet. Le mot clé void signifie que la méthode main ne retourne rien. Ne vous préoccupez pas de ces deux mots clés pour l'instant, sachez qu'ils doivent figurer devant toute méthode main invoquée au démarrage de votre application. Le tableau args contiendra tous les arguments que vous aurez passés à la ligne de commande.

Par exemple si vous appelez votre programme via la ligne de commande suivante : java Test argument1 argument2, le tableau args sera de cette forme :

```
args[0] = &quot;argument1&quot;;args[1] = &quot;argument2&quot;;
```

etc...

Source : <http://www.vulgarisation-informatique.com/classes-java.php>.

Distribution interdite sans accord écrit d'Anthony ROSSETTO (<http://www.vulgarisation-informatique.com/contact.php>)